

**Funktion und Anwendung der
Protokoll-Bibliothek Modbus RTU / Slave
(ModbusSlave.LIB)**

Dokument Version 1.0

INHALT

| | | |
|----------|------------------------------------------------|----------|
| 1 | ALLGEMEINE VORBEMERKUNGEN | 3 |
| 2 | KURZBESCHREIBUNG | 4 |
| 3 | VERWENDUNG IN EINEM PROJEKT | 5 |
| 3.1 | Neues CoDeSys Projekt anlegen | 5 |
| 3.2 | Einbindung der erforderlichen Bibliotheken | 5 |
| 3.3 | Verwendung der Bausteininstanzen in PLC_PRG | 5 |
| 3.4 | Parametrierung der Bausteininstanzen ModSlave1 | 6 |
| 4 | ÜBERSETZEN UND ABLAUF DES PROGRAMMS | 7 |
| 5 | UNTERSTÜTZTE MODBUS FUNKTIONEN | 8 |

1 Allgemeine Vorbemerkungen

Auf Grundlage der vom CoDeSys Laufzeitsystem zur Verfügung gestellten Systembibliothek SysLibCom.lib für die Verwendung der seriellen Schnittstelle wurde eine Protokoll-Bibliothek für den Datenaustausch mittels Modbus RTU entwickelt.

Diese Bibliothek ermöglicht, bei sehr einfacher Konfiguration und Parametrierung, den Datenaustausch zwischen einem Modbus-Master und einem oder mehreren Modbus-Slaves.

Diese Dokumentation beschreibt die in der Bibliothek enthaltenen Bausteine, ihre Funktion und Anwendung.

2 Kurzbeschreibung

Nachfolgend soll die Realisierung stichpunktartig erklärt werden:

Die Bibliothek besteht aus zwei Anwender-Funktionsbausteinen, die je nach verwendetem Zielsystem alternativ verwendet werden müssen. Weiterhin enthält die Bibliothek drei Datenstrukturen, die nur innerhalb des Anwender Funktionsbausteins verwendet werden.

Der Anwender benutzt je nach verwendetem Zielsystem einen der beiden Bausteine. Die Unterscheidung ist erforderlich weil je nach verwendeter Plattform die empfangenen WORD getauscht (geswappt) werden müssen oder nicht.

Für Systeme mit Motorola Byte-Order heißt der Anwenderbaustein „ModbusSlaveMSB“. Für Systeme mit Intel Byte-Order heißt der Anwenderbaustein „ModbusSlaveLSB“. Die Parameter und die Verwendung der Bausteine ist jedoch für beider Systemarten gleich.

Funktionsbausteine, die vom Anwender programmiert werden müssen:

| Nr | Art | Typ | Name | Aufgabe |
|----|----------|-----|-------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1 | Anwender | FB | ModbusSlaveLSB Alternativ: ModbusSlaveMSB | Grundbaustein für die Parametrierung einer Seriellschnittstelle und für die Abarbeitung des Modbus-Protokolls. Die von der Steuerung zur Verfügung gestellten Daten werden dem Baustein als Pointer auf die Projektdaten übergeben. |

Zum Zugriff auf die serielle Schnittstelle bedient sich die Protokoll Bibliothek der CoDeSys Systembibliothek „SysLibCom.Lib“. Diese stellt die laufzeitsystem-unabhängige Abstraktion für die Verwendung der seriellen Schnittstelle dar. Zusätzlich wird die Bibliothek „Standad.Lib“ angezogen. Beide Bibliotheken werden beim Laden der Protokoll Bibliothek automatisch eingebunden.

3 Verwendung in einem Projekt

Die Verwendung der Protokoll-Bibliothek ist sehr einfach. Das Vorgehen zur Verwendung in einem CoDeSys Projekt wird nachfolgend genau beschrieben. Dabei wird davon ausgegangen, dass ein neues Projekt angelegt werden soll.

3.1 Neues CoDeSys Projekt anlegen

- CoDeSys starten
- Mit **Datei/Neu** ein neues Projekt anlegen
- In der Zielsystem Einstellung das richtige Zielsystem auswählen, z.B. 'CoDeSys SP for Windows NT Realtime'.
- Einen neuen Baustein „PLC_PRG“ anlegen (Programmiersprache, im Beispiel „FUB“, wählen)

3.2 Einbindung der erforderlichen Bibliotheken

- Über das Menü **Fenster/Bibliotheksverwaltung** die Bibliotheksverwaltung öffnen
- Über das Menü **Einfügen/weitere Bibliothek...** die Bibliothek ModbusSlave.LIB einbinden.
- Die Bibliotheken STANDARD.LIB und SYSLIBCOM.LIB werden automatisch eingebunden. Wenn nicht, vorherigen Schritt für diese beiden Bibliotheken wiederholen.

3.3 Verwendung der Bausteininstanzen in PLC_PRG

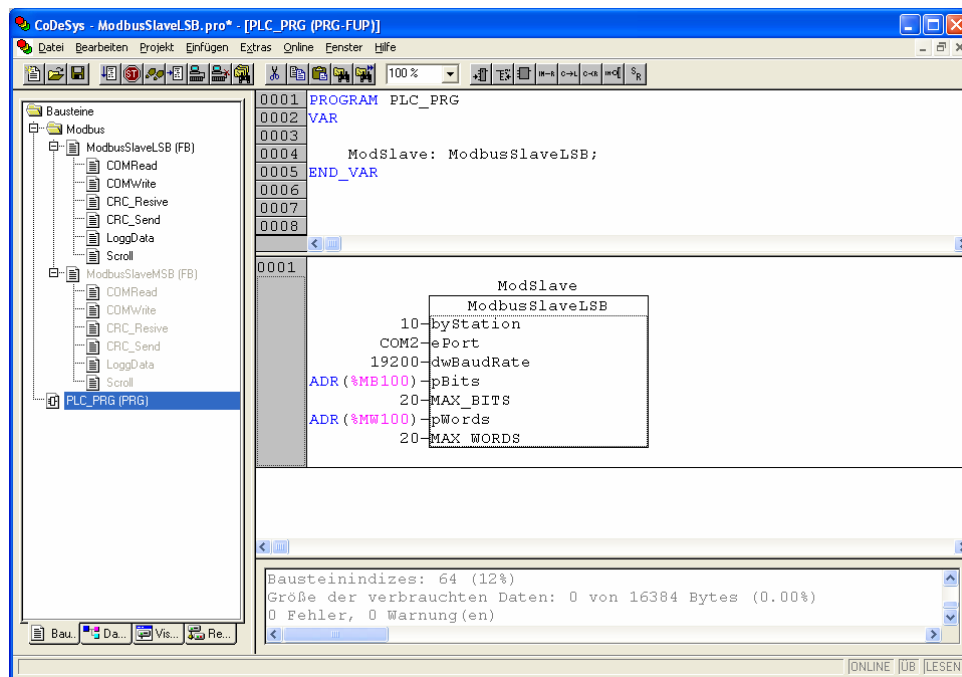
- Zur zyklischen Abarbeitung der Modbus-Slave Funktion muss innerhalb des Hauptbausteins eine Instanz des Modbus-Slave Bausteins aufgerufen werden. Hierzu wird ein Bausteine in PLC_PRG eingefügt und parametrier:
- Den Cursor in das erste Netzwerk des PLC_PRG Bausteins hinter die „???“ setzen und über das Menü **Einfügen/Baustein** einen Baustein einfügen.
- Das Schlüsselwort „AND“ innerhalb des Bausteins durch „ModbusSlaveLSB“ bzw. „ModbusSlaveMSB“ ersetzen. Als Instanznamen über dem Baustein die „???“ z.B. durch „ModSlave1“ ersetzen.
- Beim Verlassen des Eingabefeldes für den Instanznamen wird von CoDeSys automatisch der Variablendeklarations-Dialog geöffnet, der wie folgt aussieht:

- In unserem Beispiel gehen wir davon aus, dass die Instanz als lokale Variable innerhalb von PLC_PRG angelegt werden soll. Dann kann der Dialog einfach mit [OK] bestätigt werden.

3.4 Parametrierung der Baueinstanzen ModSlave1

Zur richtigen Abarbeitung muss der Baustein mit einigen Parametern versehen werden.

| Eingang | Bedeutung | Beispiel |
|------------|----------------------------------------------------------------------------------------------------------------------------------|-------------|
| byStation | Parameter für die Stationsadresse des Slaves. zulässig sind hier 1..128 | 1 |
| ePort | Parameter für die Schnittstelle zulässig sind je nach Zielsystem COM1...COM8 | COM2 |
| dwBaudRate | Verwendete Baudrate | 19200 |
| pBits | Adresse des Datenbereichs für die Modbus Bits. Im Beispiel wurden die Merker %MB100 bis %MB119 verwendet | ADR(%MB100) |
| MAX_BITS | Anzahl der BYTE, die dem Modbus-Slave als Bits zur Verfügung stehen. Im Beispiel wurden 20 Byte = 160 Bits parametrieret. | 20 |
| pWords | Adresse des Datenbereichs für die Modbus Words. Im Beispiel wurden die Merker %MW100 bis %MW119 verwendet | ADR(%MW100) |
| MAX_WORDS | Anzahl der WORDS, die dem Modbus-Slave zur Verfügung stehen. | 20 |



4 Übersetzen und Ablauf des Programms

Nach der Parametrierung des Funktionsbaustein kann das Programm über Menü **Projekt/Übersetzen** übersetzt und anschließend über **Online/Einloggen** zur Steuerung übertragen und dort mit **Online/Start** gestartet werden.

Die Bausteine sorgen für einen automatischen Datenaustausch zwischen dem Modbus-Master und der Steuerung. Hierbei werden die Daten aus dem parametrierten Datenbereichen entweder vom angeschlossenen Modbus-Master gelesen oder geschrieben. Anstelle der im Beispiel verwendeten Merker können natürlich auch eigene Datenstrukturen Angelegt werden, deren Adressen dann dem ModbusSlaveMSB/LSB Baustein übergeben werden.

5 Unterstützte Modbus Funktionen

Die Modbus-Slave Bibliothek unterstützt folgende Funktionen:

| Code | Beschreibung |
|-------|--------------------------------------------------------------------------------------------------------------------|
| 1 / 2 | Read Bits - gleichzeitiges Lesen mehrerer Bits als WORD. |
| 3 / 4 | Read Register - gleichzeitiges Lesen von einem oder mehreren WORD's mit direkt aufeinander folgenden Adressen |
| 5 | Write Bits - schreibt ein einzelnes Bit |
| 6 | Write Register - schreibt ein einzelnes WORD |
| 16 | Write Register - gleichzeitiges Schreiben von einem oder mehreren WORD's mit direkt aufeinander folgenden Adressen |

Änderungshistorie

| Version | | Freigegeben durch | Datum |
|---------|--------------------|-------------------|------------|
| V0.1 | Vorläufige Version | | 01.02.2005 |
| V0.1 | Review o.k. | AF | 09.09.2005 |
| V1.0 | Release | MN | 12.09.2005 |